Herramientas computacionales:

el arte de la analítica

(Gpo 61)

Actividad Evaluable Obtención de Estadísticas descriptivas

Link: https://github.com/J3z4r3l/

José Jezarel Sánchez Mijares | A01735226

28/ Marzo / 2023

Actividad 1 (Mapeo de coordenadas)

Objetivo:

Crear un nuevo repositorio nuevo con el nombre: Actividad 1 (Mapeo)

Implementar el código requerido para generar un mapeo del siguiente sistema global al sistema local de referencia del robot móvil y viceversa.

Introducción:

El mapeo de trayectorias es una técnica utilizada en áreas como la robótica o ingeniería de control, ya que permite generar y planificar rutas de movimiento para un objeto o agente. Esta técnica se basa en la creación de “mapas” que describen el entorno y las posibles trayectorias en este, lo que permite a su vez crear rutas más eficientes. Este proceso involucra la recopilación y procesamiento de datos de sensores y la aplicación de algoritmos de planificación de rutas para optimizar el rendimiento y garantizar la seguridad en diversas aplicaciones. En general, el mapeo de trayectorias es una herramienta importante en la creación de sistemas autónomos y en la automatización de procesos complejos.

Desarrollo:

Declaramos las variables simbólicas que nos permitirán manipular el sistema, en esta sección incluimos la matriz inercial con la que compararemos nuestro sistema local

```
%Limpieza de pantalla

clear all

close all

clc

tic

%Declaración de variables simbólicas
```

```
syms x(t) y(t) th(t)  t  %Grados de Libertad del robot móvil

%Creamos el vector de posición

xi_inercial= [x; y; th];

disp('Coordenadas generalizadas');

pretty (xi_inercial);
```

```
/  x(t) \
|       |
|  y(t) |
|       |
\ th(t) /
```

Aplicamos la derivada a la posición para obtener la velocidades generalizadas, que nos ayudará con el posicionamiento del robot

```
%Creamos el vector de velocidades

 xip_inercial= diff(xi_inercial, t);

 disp('Velocidades generalizadas');

pretty (xip_inercial);
```

```
/   d      \
|  -- x(t) |
|  dt      |
|          |
|   d      |
|  -- y(t) |
|  dt      |
|          |
|  d       |
| -- th(t) |
\ dt       /
```

Para poder determinar la posición y la rotación de nuestro objeto necesitamos crear el vector de posición que será en x y con rotación en x ya que este modelo no cuenta con desplazamiento en z, después creamos la matriz de rotación que describe el movimiento, pero falta una cosa más, ¿cómo relacionamos nuestro nuevo sistema local con el de inercia?, para lograrlo realizamos la transformación que nos permita visualizar el su comportamiento, en este caso analizamos desde el punto 0 o con respecto al mundo

```
%Defino mi vector de posición y matriz de rotación

P(:,:,1)= [x;y;th]; %Viene siendo "xi_inercial"

%Matriz de rotación alrededor del eje z....

R(:,:,1)= [cos(th) -sin(th)  0;

          sin(th)  cos(th)  0;

          0        0        1];

Realizo mi transformación del marco de referencia global al local....

xi_local=R(:,:,1)*P(:,:,1)
```

xi_local =

$$\begin{pmatrix} \cos(\mathrm{th}(t))\, x(t) - \sin(\mathrm{th}(t))\, y(t) \\ \cos(\mathrm{th}(t))\, y(t) + \sin(\mathrm{th}(t))\, x(t) \\ \mathrm{th}(t) \end{pmatrix}$$

Ahora creamos el objeto y su desplazamiento con su marco de referencia correspondiente para ello usamos una transformación que nos permitirá ver el posicionamiento de nuestro dispositivo

```
%Defino coordenadas inerciales para un tiempo 1

x1 = 1;     % Posicion inicial eje x

y1 = 1;    % Posicion inicial eje y

th1= 45;    % Orientacion inicial del robot

%Defino mi vector de posición y matriz de rotación para un tiempo 1

Pos_1=[x1; y1; th1];

Rot_1= [cos(th1) -sin(th1) 0;

       sin(th1)  cos(th1) 0;

       0         0      1];

%Realizo mi transformación del marco de referencia inercial al
local....

xi_local_1=Rot_1*Pos_1

%Obtengo la magnitud del vector resultante
```

```
magnitud= sqrt(xi_local_1(1)^2 + xi_local_1(2)^2)
inv_Rot_1= inv(Rot_1);
xi_inercial_1= inv_Rot_1*xi_local_1
```

Resultados

Como el código es el mismo para los demás incisos solo se sustituye los tres primeros parámetros de la última parte obteniendo los siguientes resultados

A. (-5,9,-2)

```
xi_local_1 = 3×1
  10.2644
   0.8012
  -2.0000
magnitud = 10.2956
xi_inercial_1 = 3×1
  -5.0000
   9.0000
  -2.0000
```

B. (-3,8,63)

```
xi_local_1 = 3×1
  -4.2965
   7.3851
  63.0000
magnitud = 8.5440
xi_inercial_1 = 3×1
    -3
     8
    63
```

C. (5,-2,90)

```
xi_local_1 = 3×1
  -0.4524
```

```
   5.3661
  90.0000
magnitud = 5.3852
xi_inercial_1 = 3×1
   5.0000
  -2.0000
  90.0000
```

D. (0,0,180)

```
xi_local_1 = 3×1
     0
     0
   180
magnitud = 0
xi_inercial_1 = 3×1
     0
     0
   180
```

E. (-6,3,-55)

```
xi_local_1 = 3×1
  -3.1320
  -5.9322
 -55.0000
magnitud = 6.7082
xi_inercial_1 = 3×1
  -6.0000
   3.0000
 -55.0000
```

Conclusión

Como ya se mencionó durante esta actividad se realizó un proceso en el que se utilizaba el mapeo de coordenadas en este caso en particular, se llevó a cabo un análisis detallado para visualizar el comportamiento de un robot que se mueve desde sus coordenadas iniciales en un plano X,Y. en la mayoría de los casos se observa como los resultados obtenidos de la magnitud que podríamos establecer como el desplazamiento varían dependiendo a los parámetros dados, en el caso d) se observa que no se desplazó, esto se debe a que solo roto, ya que, al no tener desplazamiento en z sólo puede rotar sobre ese eje.

En resumen, el mapeo de coordenadas es una herramienta clave para la navegación autónoma de robots en un plano 2D, y el análisis detallado de su comportamiento es esencial para planificar y trazar trayectorias de manera precisa y evitar obstáculos con éxito.